

Evaluating the use of Pareto Efficiency to Optimize Non-Functional Requirements Satisfaction in i^* Modeling

J. Zubcoff, I. Garrigós, S. Casteleyn, J.N. Mazón, J.A. Aguilar

Abstract—Due to the large, heterogeneous audience of Web applications, and its rapidly changing expectations, holistic requirement analysis approaches are crucial to ensure the success of Web engineering projects. To increase the quality of resulting Web applications, non-functional requirements (NFRs) must be considered. Satisfying them is a non-trivial task that depends on making decisions about which functional requirements (FRs) to implement, and how to prioritize the NFRs. A satisfactory solution is a trade-off, where competing NFRs must be balanced. In this paper, we outline how the Pareto efficiency can complement a goal-oriented requirement analysis modelling to evaluate and select optimal configurations of requirements for a Web application, while NFRs are balanced and maximized according to a priority list. We hereby focus on an empirical evaluation to verify whether our Pareto method improves the accuracy of design decisions during the requirements analysis phase, and/or if it reduces the time needed by designers.

Index Terms—Pareto Efficiency, Web Engineering, NFRs Optimization, i^* , goal-oriented requirements.

I. INTRODUCCIÓN

Diferentes estudios han demostrado que un análisis de requisitos efectivo es un factor de éxito crítico en los proyectos de software, ej. [1]. En general, se diferencian dos tipos de requisitos: Requisitos Funcionales y Requisitos No-Funcionales (FRs y NFRs, por sus siglas en inglés). Los primeros describen la funcionalidad del producto a desarrollar, mientras que los segundos (también conocidos como requisitos “de calidad”) permiten especificar restricciones en el producto, en el proceso de desarrollo [2] así como mejorar significativamente la satisfacción del usuario. Con este fin, deben de ser considerados desde las etapas tempranas del proceso de desarrollo, es decir, desde la etapa de análisis de requisitos [3].

En Ingeniería Web, la etapa de análisis de requisitos resulta particularmente difícil debido a la gran y heterogénea audiencia de este tipo de aplicaciones, la rápida evolución/cambio de los requisitos, así como a la relación de

NFRs, ej. que FR es necesario implementar para maximizar la usabilidad, por consiguiente, son necesarios métodos específicos para el análisis de requisitos en Ingeniería Web (evaluados empíricamente) que permitan al diseñador considerar, además de los FRs, a los NFRs desde las etapas iniciales del proceso de desarrollo, con la finalidad de satisfacer la experiencia de navegación del usuario [4].

El modelado conceptual es la base del análisis de requisitos, su propósito es apoyar la comprensión del dominio del problema mediante la representación de las entidades del mundo real, sus relaciones y dependencias. Específicamente, las aproximaciones para el modelado de requisitos orientado a objetivos (*goal-oriented requirements*) son utilizadas para analizar el dominio del problema y explícitamente representar a los FRs, los NFRs y las relaciones que existen entre ellos. Tales enfoques (ej. [5]) soportan el razonamiento sobre varias alternativas de diseño en base a los FRs (representados como tareas) y evalúan los efectos de estos sobre los NFRs (representados como *softgoals*). De este modo, los diseñadores tienen información útil para decidir qué FRs deben de ser implementados y cuáles no deberían serlo. Existen diversos escenarios en donde este tipo de enfoques son necesarios ej. cuando las restricciones de tiempo y de costo a menudo impiden la implementación de todos los FRs [6]. En el contexto de la Ingeniería Web, los requisitos tienden a evolucionar rápidamente [7], por lo cual es importante ser capaz de evaluar el impacto de nuevos requisitos o de la evolución de los existentes. En un proceso de desarrollo, en una iteración incremental sólo una colección de FRs se implementará en la aplicación Web final. Por lo tanto, en todos estos escenarios, es indispensable un mecanismo que permita al diseñador evaluar el impacto de la implementación de los FRs en base a los NFRs.

A pesar de que los enfoques de modelado orientado a objetivos ofrecen modelos gráficos para representar FRs, NFRs y sus relaciones (ej. el marco de modelado i^* [8] y su adaptación al dominio Web [9]), desde un punto de vista práctico, nuestra hipótesis es que aun así es difícil para los diseñadores interpretar correctamente las interacciones complejas que existen en un modelo i^* y decidir cuales FRs implementar para obtener un balance satisfactorio de NFRs de acuerdo a su prioridad (ej. la optimización de NFRs). Por esa razón, en nuestro trabajo previo [10] presentamos un método basado en la eficiencia de Pareto, con el soporte de la herramienta WebREd [11], que complementa los modelos

J. Zubcoff, Universidad de Alicante, Alicante, Spain, jose.zubcoff@ua.es.
 I. Garrigós, Universidad de Alicante, Alicante, Spain, igarrigos@ua.es.
 J.N. Mazón, Universidad de Alicante, Alicante, Spain, jnmazon@ua.es.
 S. Casteleyn, Vrije Universiteit Brussel, Belgium, sven.casteleyn@vub.ac.be.
 J.A. Aguilar, Universidad Autónoma de Sinaloa, Mazatlán, México, ja.aguilar@uas.edu.mx.

compensación (trade-off) y optimización entre los FRs y los

conceptuales orientados a objetivos, lo que ayuda a los diseñadores a tomar decisiones mejor fundamentadas mediante la comprensión de la relación de compensación (trade-off) necesaria para obtener NFRs optimizados y equilibrados, de esta forma les permite comparar fácilmente diferentes configuraciones de FRs [12]. En este contexto, la eficiencia de Pareto es particularmente útil, porque se desempeña bien cuando deben equilibrarse múltiples objetivos competitivos y conflictivos (es decir, la satisfacción de varios NFRs al mismo tiempo, eje. la seguridad frente a la usabilidad). La eficiencia de Pareto, define cómo generar la frontera de Pareto, formada por un listado de configuraciones de FRs, en donde ningún NFR se puede optimizar sin afectar a otro NFR, lo que proporciona al diseñador un conjunto de soluciones perfectamente equilibradas entre las cuales se puede elegir la solución final tomando como base las prioridades establecidas sobre los NFRs, es decir, cual NFR es necesario optimizar.

En este artículo, complementamos nuestro método para el modelado de requisitos Web con la eficiencia de Pareto. Nos enfocamos en una evaluación empírica utilizando la siguiente hipótesis: ¿puede el método de eficiencia de Pareto apoyar eficazmente a los diseñadores de aplicaciones Web al momento de evaluar mejor las diferentes configuraciones de requisitos (FRs y NFRs)? Por tanto, analizaremos si nuestro método permite a los desarrolladores seleccionar una configuración óptima y equilibrada (entre FRs y NFRs) con más precisión y de una manera más rápida, para establecer qué FRs es necesario implementar en la aplicación Web final para conservar un balance entre los NFRs.

II. UN ENFOQUE ORIENTADO A OBJETIVOS PARA LA ESPECIFICACIÓN DE REQUISITOS WEB

Esta sección introduce nuestra propuesta orientada objetivos para el análisis de requisitos Web a través de modelos i^* [13]. El marco de trabajo i^* está constituido por dos modelos: el modelo de dependencia estratégica (SD por sus siglas en inglés), utilizado para describir las relaciones de dependencia entre varios actores en un contexto organizacional, representado por (\neg) y el modelo estratégico racional (SR, por sus siglas en inglés) utilizado para describir los intereses y preocupaciones del actor (\bigcirc) y como es que se podrían abordar, su símbolo es (\odot) . El modelo SR provee una forma detallada de modelar elementos intencionales internos y las relaciones de cada actor. Los elementos intencionales son objetivos (*goals*, \bigcirc), tareas (*tasks*, \hexagon), recursos (*resources*, \square) y *softgoals* (∞) relacionados entre sí por las denominadas relaciones intencionales, que son: enlaces (*links*) medio-fín (*means-end*, \neg) para representar formas alternativas de cumplir con un objetivo; enlaces de descomposición de tareas (*task-decomposition*, \dashv), utilizados para relacionar los sub-componentes necesarios para que se pueda llevar a cabo una tarea y los enlaces de contribución (*contribution links*, \Rightarrow) para modelar como es que un elemento intencional contribuye a la satisfacción o

cumplimiento de una *softgoal*. Los enlaces de contribución están clasificados en diferentes tipos, estos son: “*Make*”, indica la satisfacción de una *softgoal*, “*Some+*” para mostrar una contribución positiva fuerte y “*Help*” para señalar una de menor fuerza; sus homólogos negativos son “*Break*”, “*Some-*”, “*Hurt*”. Finalmente, la etiqueta “*Unknown*” indica un tipo de contribución cuya fuerza se desconoce.

Para adaptar el marco de modelado i^* al dominio de la Ingeniería Web utilizamos la clasificación de requisitos Web presentada en [14], de tal forma que el elemento “tarea” de i^* , que se utiliza para modelar a los FRs, se utilizó para representar requisitos específicos para la Web: servicio (*service*, S), navegacional (*navigational*, N), interfaz (*layout*, L) y requisitos de personalización (*personalization*, P). Los requisitos de contenido (*content*, C) se definen como una especialización del elemento recurso (*resource*) de i^* y los NFRs se modelan directamente con el elemento *softgoal* (Utilizaremos NFRs en situaciones generales, mientras que el término *softgoal* será aplicado cuando se describa un modelo i^*) del marco de trabajo i^* . A continuación, se utilizaron los mecanismos de extensión de MOF (*Meta Object Facility*, por su siglas en inglés) para definir un metamodelo para el uso de i^* con esta terminología, por lo que fueron agregadas nuevas metaclases para cada uno de los diferentes tipos de requisitos Web.

Un ejemplo de la aplicación del marco de modelado i^* para el dominio Web se muestra en la Fig. 1. El objetivo del sistema es brindar soporte en la realización de una encuesta en línea. La aplicación Web se explicará con más detalle en la Sección 4.

III. OPTIMIZACIÓN DE NFRs EN EL MODELADO DE REQUISITOS WEB

En esta sección presentamos un método para la optimización de NFRs basado en la eficiencia de Pareto. El método asiste a los diseñadores en la evaluación y selección de los FRs que se deberán o no implementar en la aplicación Web a desarrollar con finalidad de aumentar la calidad con respecto a los NFRs. El propósito es brindar soporte y mejorar el análisis de requisitos con la eficiencia de Pareto mediante la obtención de un conjunto de configuraciones de FRs óptimas, que se encuentren balanceadas y equilibradas considerando la maximización del NFR.

La eficiencia de Pareto, es una noción de economía aplicada en ingeniería que es útil cuando hay múltiples objetivos contradictorios y conflictivos que deben ser equilibrados [12], se puede describir de la siguiente manera: “dado un conjunto de individuos, un conjunto de asignaciones alternativas y un conjunto de valoraciones dependientes de asignación, una asignación A es una mejora sobre la asignación B si A puede hacer al menos una valoración mejor que B, sin hacer ninguna otra peor”. Intuitivamente, la frontera de Pareto es el conjunto de asignaciones que no pueden ser mejoradas de manera uniforme. Aplicando este principio a nuestro dominio, el conjunto de individuos se refiere al conjunto de los FRs que se deberán implementar en la aplicación Web, las asignaciones

alternativas corresponden con el conjunto de FRs en un cierto estado (implementado o no implementado), es decir, una configuración, y la mejora de la valoración particular, se refiere a una mejor satisfacción de un NFR particular. Por lo tanto, una configuración de la frontera de Pareto es una asignación de estados para los FRs (implementado o no implementado) de forma que ninguna otra configuración satisface mejor a un solo NFR, al tiempo que satisface los otros por igual. La frontera de Pareto suele contener varias configuraciones, cada una representando un equilibrio óptimo entre los NFRs. Mientras que un NFR puede ser satisfecho positiva o negativamente en las diferentes configuraciones de la frontera de Pareto, estamos seguros de que ninguna de las configuraciones de la frontera de Pareto pueden ser modificadas de ninguna manera sin afectar negativamente al menos a un NFR. La frontera de Pareto, por lo tanto, puede ser utilizada por el diseñador para evaluar el impacto de la implementación de los FRs sobre cada uno de los NFRs, con lo que podrá tomar una decisión bien informada sobre cuál de todas las configuraciones mejor satisface a los NFRs, según las prioridades establecidas por los *stakeholders*.

Con la finalidad de brindar soporte a cada paso de este proceso, se ha implementado una herramienta (WebRED), constituida por una serie de *plugins* de Eclipse. La herramienta combina un conjunto de tecnologías como EMF (*Eclipse Modeling Framework*), GMF (*Graphical Modeling Framework*) y J2EE, utilizadas para implementar el metamodelo para requisitos Web, el ambiente gráfico de la herramienta de modelado de requisitos y el algoritmo de eficiencia de Pareto respectivamente. Una descripción detallada de WebRED se puede encontrar en [11].

IV. EVALUACIÓN EXPERIMENTAL

En esta sección se describe el contenido principal de este artículo: una evaluación empírica conducida para calcular la eficiencia y eficacia de nuestro enfoque basado en Pareto para el análisis de requisitos Web. Para llevar a cabo el experimento, se emplearon seis pasos descritos a continuación:

Paso 1. Preparación del Experimento: en este paso se establecieron los objetivos generales del experimento: (i) probar la eficiencia (tiempo) y efectividad (exactitud de resultados) de las representaciones gráficas para la maximización de NFRs en Ingeniería Web (modelos/diagramas *i** y la frontera de Pareto) y (ii) probar la complejidad en ambos mecanismos. Para lograr estos objetivos, se creó un experimento con dos factores fijos: tipo de representación (factor tipo, es decir, modelo *i** o frontera de Pareto) y la complejidad (factor de complejidad, compuesto por dos niveles: fácil o complejo). Había dos variables dependientes: el tiempo (en segundos) y la puntuación obtenida por los individuos (osciló de 0 a 2). Estas variables se miden mediante el uso de tiempo de respuesta de los sujetos en el cumplimiento de las tareas requeridas (tiempo) y al juzgar la exactitud de sus respuestas (score). Por consiguiente, podemos medir la eficiencia y la eficacia tanto de los modelos

*i** y las representaciones de la frontera de Pareto utilizando las variables de tiempo y de puntuación.

Paso 2. Preparación del Material: En nuestro experimento, hemos definido cuatro ejercicios (tres preguntas cada uno) de acorde al nivel de complejidad de los modelos (fácil o complejo) y al tipo de representación visual de la información (tipo: diagrama/modelo *i** o frontera de Pareto). En la Fig. 1 se puede observar un ejemplo de un modelo *i** de nivel de complejidad “fácil”, el objetivo (*goal*) de este modelo es realizar una encuesta (“Survey to be performed”) y este puede ser alcanzado a través de alguno de los siguientes requisitos navegacionales:

“Interactive Interview”: este requisito beneficia la usabilidad (*usability*), pero perjudica la confiabilidad (*reliability*). Se puede cumplir si los requisitos navegacionales “Establish chat connection” y “Perform interview” son alcanzados, para esto, es necesario el requisito de contenido “Survey repository”.

“Public Questionnaire”: este requisito realiza una contribución positiva al NFR anonimato (*anonymity*) y perjudica a la fiabilidad (*reliability*). Se puede cumplir si el requisito navegacional “Perform survey” y el requisito de servicio “Send survey” son alcanzados, es necesario el requisito de contenido “Survey repository”.

“Private Questionnaire”: realiza una contribución positiva a la fiabilidad (*reliability*), una negativa al NFR anonimato (*anonymity*) y afecta también de forma negativa a la usabilidad (*usability*). Puede ser alcanzado si el requisito navegacional “Perform private survey” es logrado, es necesario cumplir con los requisitos de servicio “Login” y “Send private survey”.

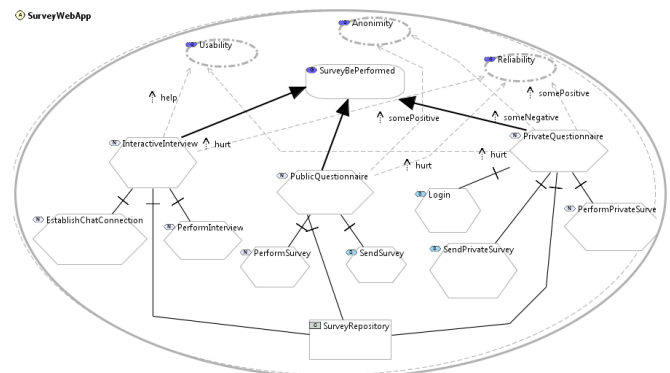


Fig. 1. Modelo *i** de complejidad “fácil” y de tipo “modelo *i**”.

Posteriormente, se elaboraron tres preguntas, la primera de ellas se relaciona a un solo NFR, mientras que la segunda y tercera pregunta se refieren a dos y tres NFRs con diferentes prioridades. Las preguntas son las siguientes:

Pregunta 1: ¿Qué tareas se necesitan implementar para maximizar la usabilidad (*usability*)?

Pregunta 2: ¿Qué tareas se necesitan implementar para maximizar la usabilidad (*usability*) y la confiabilidad (*reliability*) al mismo tiempo (misma prioridad)?

Pregunta 3: ¿Qué tareas se necesitan implementar para maximizar la usabilidad (*usability*) que es la de mayor

prioridad) y después la confiabilidad (*reliability*) y el anonimato (*anonymity*)?

En la figura 2 se observa un ejemplo de un modelo i* de nivel de complejidad “fácil” y de tipo “frontera de Pareto” acerca de una aplicación Web para pago en línea. En este ejemplo se pueden observar siete columnas, la primera columna se utiliza para representar el identificador de cada configuración (filas de la tabla) que se genera en la frontera de Pareto, las siguientes tres columnas son FRs o requisitos Web (R0: offlinepayment, R1: onlinepayment, R2: callbackpayment) y tres son NFRs (Usability, Responsiveness y Security). Para cada configuración, la letra “I” en la celda significa que el FR se debe implementar y la letra “N” significa que el requisito no se deberá implementar. Las celdas que contienen valores numéricos corresponden con la suma de todos los enlaces de contribución para ese NFR en esa configuración, en donde “Make” aporta un valor de “+4”, “Some+” contribuye con “+2” y “Help” con un valor de “+1”, de forma similar para los valores negativos (“Break”, “Some-”, “Hurt”). La explicación completa del experimento se encuentra disponible en la dirección: <http://webred.maz.uasnet.mx/descargas/ieeepaperpareto.zip> para consulta o reproducción de forma libre.

Config.	R0	R1	R2	Usability	Responsiveness	Security
X2	I	I	N	1	-2	0
X3	I	N	I	-1	-3	1
X4	I	N	N	0	-4	1
X5	N	I	I	0	3	-1
X6	N	I	N	1	2	-1
X7	N	N	I	-1	1	0
X8	N	N	N	0	0	0

Fig. 2. Ejemplo de un modelo i* de complejidad “fácil” y de tipo “frontera de Pareto i*”.

Se utilizaron modelos diferentes para representar gráficamente modelos i* (diagrama) y frontera de Pareto (tabla), se aseguró que para cada nivel de complejidad, los modelos subyacentes para ambas representaciones gráficas (tipo) tuvieran exactamente el mismo nivel de dificultad. Para lograr esto, se utilizaron la misma cantidad de objetivos, NFRs, tareas/requisitos Web, recursos, enlaces de contribución y de medio-fin así como la misma jerarquía de tareas. Además, se especificaron la misma cantidad de enlaces de contribución tareas- *softgoals* positivos y negativos, y la misma cantidad para los enlaces de contribución del tipo *softgoal*-tarea.

Paso 3. Definición del contexto: en este paso se identificaron los sujetos que participarían en el experimento, estudiantes del Máster Oficial de tecnologías Web de la Universidad de Alicante (España). Por una parte, los sujetos son expertos desarrollo Web y a su vez tenían conocimiento previo del modelado de requisitos con i* (un curso de 30 horas). Sin embargo, no sabían nada acerca de la eficiencia de Pareto, por lo que fue necesario llevar a cabo una sesión de capacitación para proporcionar los conocimientos necesarios en el experimento.

Paso 4. Conducción del Experimento: Para cada ejercicio (cuatro en total), los participantes escribieron la respuesta a las

preguntas así como la hora de inicio y final de cada respuesta en horas, minutos y segundos.

Paso 5. Pre-procesamiento de la Información: la variable tiempo mide el tiempo que necesitan los sujetos de prueba para responder cada pregunta individual, se obtuvo directamente restando el tiempo inicial menos el tiempo final. Posteriormente se utilizó una puntuación para medir la exactitud de la respuesta, el rango de la puntuación fue de 0 a 2 (0 para una respuesta incorrecta; 1 para una respuesta parcialmente correcta; 2 para una respuesta completamente correcta).

Paso 6. Análisis de los Datos: en este paso se utilizó el Análisis de la Varianza [14] (ANOVA) debido a que permite explorar los resultados de un diseño factorial para una determinada variable en estudio y sirve para comparar si los valores de un conjunto de datos numéricos son significativamente distintos a los valores de otro o más conjuntos de datos. ANOVA se utiliza para asociar una probabilidad a la conclusión de que la media de un grupo de puntuaciones es distinta de la media del otro grupo. En primer lugar, fueron examinadas la homogeneidad de las varianzas y la normalidad de los datos, posteriormente se realizó un análisis descriptivo para obtener información adicional sobre las variables dependientes (como la media y la desviación estándar de la puntuación y el tiempo) y su comportamiento a lo largo de los diferentes niveles de los factores. Después, se realizó una prueba de ANOVA para evaluar la influencia del tipo de modelo (modelo i* o frontera de Pareto) y el nivel de complejidad (fácil o complejo) en el marcador (debido a que es la variable en estudio, la llamamos variable dependiente). El mismo análisis se realizó para saber si estos factores tienen un efecto significativo estadísticamente sobre la variable “tiempo” (la otra variable de estudio).

V. ANÁLISIS DE LOS RESULTADOS

La estadística descriptiva y puntuación para la variable dependiente se detallan en la Tabla 1. En esta tabla se considera el tipo de modelo (modelo i*: *i_star* o frontera de Pareto: *pareto_front*) y el nivel de complejidad (fácil o complejo). Los valores entre paréntesis representan la media y la desviación estándar. La media de la puntuación de la frontera de Pareto es superior a la media para el modelo i* para ambos niveles fácil y complejo. Por lo tanto, es obvio que el uso de la frontera de Pareto aumenta la exactitud de las respuestas, y por lo tanto es más eficaz para la selección de qué requisitos Web implementar en comparación con el uso de modelos i*. Observamos, además, que para el diagrama i*, la media se reduce drásticamente (casi a la mitad) para un escenario complejo, mientras que para la frontera de Pareto sólo disminuye ligeramente. Esto apunta al hecho de que la solución de diagrama i* no escala bien con el aumento de la complejidad, mientras la solución en la frontera de Pareto escala mejor.

Curiosamente, la variabilidad, representada por la desviación estándar, es alta comparada con la media en el caso de i*, lo cual se debe al hecho de que los sujetos no son

expertos en los conceptos de modelado con i^* , por lo que algunos de los ellos pudieron modelar bien y otros no. Este comportamiento puede ser también observado en el coeficiente de variación (CV) [15], que es una medida normalizada de la dispersión. El CV de las puntuaciones de i^*

TABLA I
ESTADÍSTICA DESCRIPTIVA PARA LA PUNTUACIÓN

Variable	Dificultad Fácil	Dificultad Complejo
i_{star}	1.37 (0.94)	0.70 (0.93)
$CV(i_{star})$	68.61%	132.85%
$pareto_{front}$	1.57 (0.70)	1.42 (0.85)
$CV(Pareto_{front})$	44.58%	59.85%

aumenta drásticamente desde el 68,61% para el modelo fácil a 132,85% para el modelo complejo. Sin embargo, las puntuaciones de la frontera de Pareto incrementan el CV (de 44,58% a 59,85%). Luego, la frontera de Pareto obtuvo mejores puntajes (el doble de la puntuación) que los modelos i^* y además, los resultados son más precisos (menor variabilidad) en la frontera Pareto que en los del modelo i^* .

La otra variable en estudio es el tiempo, medido en segundos, se utiliza para obtener información acerca de la eficacia de la frontera de Pareto. La Tabla 2 muestra la media

TABLA II
ESTADÍSTICA DESCRIPTIVA PARA EL TIEMPO

Variable	Dificultad Fácil	Dificultad Complejo
i_{star}	147.17 (106.99)	135.97 (84.03)
$CV(i_{star})$	72.7 %	61.8 %
$pareto_{front}$	74.47 (58.81)	98.87 (41.72)
$CV(Pareto_{front})$	78.97 %	42.2 %

y desviación estándar (este última entre paréntesis) del tiempo que los sujetos utilizaron para responder a las preguntas de cada ejercicio. El comportamiento CV es similar a los diagramas i^* y frontera de Pareto, sin embargo, la media de la variable tiempo es siempre más baja (significativamente menor, ver ANOVA en la Tabla 4) para la frontera de Pareto que para el diagrama i^* .

De la Tabla 2, se observa que el valor medio para la frontera de Pareto es para ambos niveles de complejidad más bajo (es decir, mejor) que para el diagrama de i^* , mostrando así que los sujetos realizaron sus tareas más rápido con la frontera de Pareto. Además, de la Tabla 1 llegamos a la conclusión de que los sujetos también obtuvieron puntuaciones más altas con la frontera de Pareto, por lo que el resultado global es claramente favorable para la frontera de Pareto, tanto en la eficiencia (tiempo) y eficacia (exactitud de los resultados).

Los sujetos de prueba realizan sus tareas un poco más rápido en el escenario complejo utilizando el diagrama de i^* , en comparación con el escenario fácil, pero sus puntuaciones se redujeron a la mitad. Esto podría indicar que el diagrama i^* complejo provocó desconcierto en los sujetos de prueba, lo que los llevó a formular una respuesta sin tener en cuenta todas las posibilidades, se necesita más investigación para determinar la causa exacta. Para la frontera de Pareto, el

tiempo para resolver las tareas aumentó para el escenario complejo, mientras que la puntuación sólo disminuye ligeramente, lo que indica que la frontera de Pareto permanece manejable para los sujetos de prueba, incluso en escenarios más complejos.

El resultado de la prueba de ANOVA para la variable puntuación se muestra en la Tabla 3. Este es un diseño con dos factores fijos y su interacción. Es importante evaluar la interacción entre el tipo de ejercicio y la complejidad, debido a que nos proporcionará información valiosa sobre el comportamiento de las herramientas de modelado cuando se trata de modelos sencillos o complejos. La prueba de ANOVA fue significativa para ambos factores y también para su interacción, las diferencias entre las medias para el tipo de ejercicio (modelo i^* y frontera de Pareto) fueron significativas ($p = 0,0$). Por otra parte, la complejidad tiene efecto en la puntuación ($p = 0,0003$), sin embargo, la fuente más

TABLA III
TABLA ANOVA PARA LA PUNTUACIÓN

Fuente de Variabilidad	Df	Sum Sq	Mean Sq	F Value	Pr(>F)
Type	1	12.60	12.60	17.11	0.0000
Complexity	1	10.00	10.00	13.58	0.0003
Type: Complexity	1	4.00	4.00	5.44	0.0206
Residuals	1	173.85	0.74		

importante de la variabilidad se convierte en la interacción entre los dos factores ($p = 0,0206$) lo que significa que no existen diferencias significativas cuando se utiliza la frontera de Pareto o el modelo i^* para los modelos fáciles y/o complejos. Las puntuaciones de la frontera de Pareto son significativamente mejores que la puntuación en los modelos i^* en ambos modelos fáciles y complejos (Tabla 3).

Por otra parte, los resultados obtenidos de la prueba de ANOVA para el tiempo utilizado para resolver el cuestionario muestran un comportamiento ligeramente diferente. Mientras que el tipo de modelo es significativo ($p = 0,0$), el tiempo medio para la frontera de Pareto es menor que el tiempo medio de i^* , la complejidad no afecta el tiempo (la variable de estudio). El valor de p de 0.5073 indica que no hay diferencias entre el nivel de dificultad de los ejercicios (incluyendo todos los valores de la variable tiempo, para i^* y frontera de Pareto) cuando se mide el tiempo. Esto se puede observar en la figura 5, que representa el *boxplot* para el tiempo, se deduce que hay una mayor variabilidad durante el tiempo cuando se trabaja con los modelos de nivel de complejidad fácil. La variabilidad está muy influenciada por los valores atípicos que pueden ser encontrados tanto en los modelos i^* y en la frontera de Pareto de complejidad fácil. Este comportamiento se debe a la falta de experiencia de los sujetos de estudio. Sin embargo, la interacción tiene un significado intermedio ($p = 0,0746$) que significa que hay algún efecto de interacción entre los dos factores, esto se puede observar mejor en la gráfica de interacción (Fig. 6). Como resultado, la variabilidad en el tiempo puede producir una perturbación en el análisis, pero no

oculta las diferencias entre los tiempos medios obtenidos de la frontera de Pareto (siempre es menor) y del modelo (diagrama) de i^* (es superior).

El efecto de la interacción de ambas variables dependientes se muestra en la Fig. 6. Estos son los gráficos de interacciones de las variables de puntuación y del tiempo por los niveles de tipo y complejidad. La figura 6a representa los resultados obtenidos por los individuos (cuanto más alto mejor), podemos observar cómo la frontera de Pareto obtiene valores similares (es decir, no hay diferencia significativa) de respuesta ($p = 0,77$) cuando se trata de modelos de nivel de dificultad fácil o complejo. Sin embargo, cuando se utiliza el diagrama de i^* , la puntuación obtenida por los individuos en la resolución de un ejercicio fácil fue significativamente diferente de los que tienen nivel de dificultad complejo ($p = 0,0001$). Hay una disminución de las puntuaciones cuando el tipo de modelo cambia de fácil a complejo (Fig. 6a).

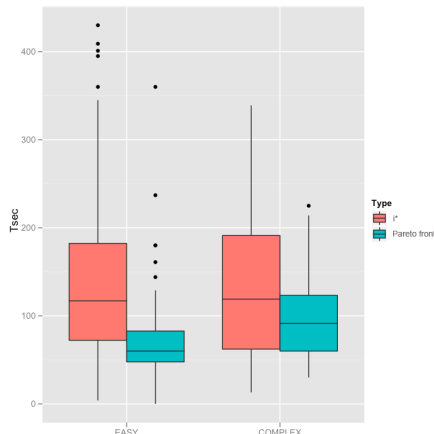


Fig. 5. Boxplot para el tiempo en segundos.

Por otro lado, no hay una disminución de las puntuaciones cuando se utiliza la frontera de Pareto (la línea en la Figura 6a). La media de las puntuaciones es bastante similar para los modelos fáciles o complejos y el comportamiento mostrado sigue siendo similar en modelos fáciles o complejos, lo cual es muy conveniente para los escenarios de la vida real en donde la escalabilidad de las soluciones se debe de tener en cuenta. Por el contrario, las puntuaciones del modelo i^* disminuyen casi a la mitad para los modelos complejos cuando se comparan con los resultados de los modelos fáciles.

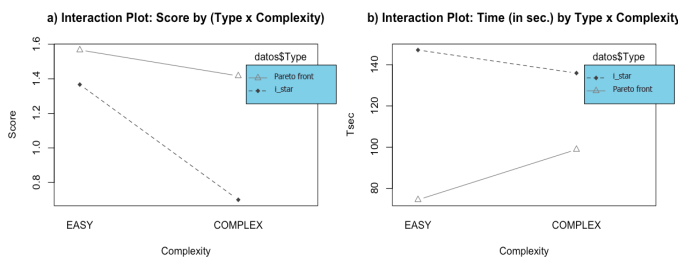


Fig. 6. Diagrama de interacción para las variables de puntuación y Tsec.

La figura 6b representa el tiempo para los niveles de tipo y complejidad (menor es mejor). El comportamiento de la frontera de Pareto fue bastante similar, la complejidad no tiene

un efecto significativo en la variable de tiempo ($p = 0,3$). En este caso, el diagrama i^* en la Fig. 6b no muestra una fuerte disminución en la variable de tiempo, sin embargo, el tiempo en segundos fue significativamente mayor que los obtenidos con el uso de la frontera de Pareto ($p = 0,0$). Es importante mencionar que tenemos un resultado inesperado para el modelo i^* : para los modelos complejos se consume menos tiempo (en promedio) que en los modelos fáciles. La revisión detallada de los datos verificó que esas respuestas rápidas eran erróneas, entonces, el rendimiento general de la frontera de Pareto muestra que es más conveniente y más eficiente que el modelo i^* , principalmente por su mejor resultado (eficacia) y menor tiempo consumido (eficiencia) en cualquier situación (modelos fáciles o complejos) en la gestión de NFRs.

VI. TRABAJO RELACIONADO

En nuestro trabajo previo [16] se realizó una revisión sistemática de la literatura para estudiar las técnicas utilizadas en Ingeniería de Requisitos (Requirements Engineering) en el desarrollo de aplicaciones Web. Los resultados obtenidos muestran que la mayoría de los enfoques para el desarrollo de aplicaciones Web se encaminan principalmente a las etapas de análisis y diseño, por lo que no ofrecen un soporte completo a la etapa de requisitos, incluso, la evaluación de estos enfoques en algunos casos es inexistente o carece de rigor científico. También, los NFRs son dejados de lado o considerados de una forma muy aislada, en algunos casos esto se debe a que resulta difícil su representación aplicando las técnicas de análisis de requisitos más utilizadas por estos enfoques: Casos de Uso y Perfiles UML (*Unified Modeling Language*, por sus siglas en inglés). Por otra parte, en lo que respecta a la consideración de los NFRs desde las etapas iniciales del proceso de desarrollo, en [5] encontramos el estado-del-arte referente a los NFRs y cómo es que son considerados en el contexto del *Model-Driven Development* (MDD, por sus siglas en inglés), desafortunadamente, esta investigación se contextualiza fuera de la Ingeniería Web, en cambio, en [17], los autores proponen un metamodelo para la representación de requisitos de usabilidad en aplicaciones Web y en [18] podemos encontrar un enfoque para analizar alternativas de diseño de forma sistemática considerando a los NFRs como punto de partida. Desafortunadamente estos trabajos pasan por alto como balancear y optimizar NFRs. Desde el punto de vista de modelado conceptual, destacamos la investigación de Moody [19], la cual indica que se requiere más investigación empírica en el modelado conceptual, debido a que la mayor parte de la investigación en esta área es de naturaleza teórica. Desde entonces, se han hecho esfuerzos en la comunidad de modelado conceptual para abordar este problema, por ejemplo, se han realizado evaluaciones empíricas para proporcionar información sobre el uso de modelos conceptuales por parte de los *stakeholder's* en la empresa [20] para la representación de las relaciones *part-whole* en el modelado conceptual [21], o para comparar diferentes lenguajes de modelado conceptual de datos [22]. En lo que respecta a la ingeniería de requisitos orientada a objetivos (*goal-oriented requirements engineering*), existen algunos trabajos que resaltan la importancia de la evaluación de este tipo de enfoques, Al

Subaie y Maibaum [23] presentan una evaluación cualitativa del método KAOS y su respectiva herramienta *Objectiver*, este tipo de evaluación es subjetiva debido a que la información cualitativa de KAOS y *Objectiver* se basa en las observaciones de los evaluadores, usaron KAOS y *Objectiver* para abordar un problema objetivo para obtener informes y observaciones con el fin de evaluar la pertinencia, la exactitud, la trazabilidad y la comprensibilidad del enfoque. En cambio, en Teruel et al [24] estudiaron la aplicabilidad de tres técnicas utilizadas en la ingeniería de requisitos (casos de uso, orientados a objetivos y *viewpoints*) para especificar sistemas colaborativos, realizaron un análisis empírico para comparar las técnicas por medio de una variación del método DESMET [25], utilizado para evaluar métodos y herramientas en ingeniería de software. En Morandini et al. [26] se presentó un informe sobre la evaluación empírica de Tropos4AS (Tropos para Sistemas Adaptativos) sobre la eficacia de los modelos de Tropos4AS, en dos experimentos los participantes que tuvieron que realizar varias tareas y dar sus opiniones personales. Por último, en Estrada et al [27] se evaluó empíricamente el uso de *i** en un entorno de generación de modelos de software en un ambiente industrial.

Recapitulando, se han realizado varios intentos para proporcionar técnicas y métodos para tratar con algunos aspectos del proceso de Ingeniería de Requisitos en aplicaciones Web, sin embargo, todavía se necesitan soluciones integrales que consideren a los NFRs desde las etapas iniciales del proceso de desarrollo para satisfacerlos de igual forma que a los FRs y por consiguiente mejorar la calidad de la aplicación Web a desarrollar.

VII. CONCLUSIONES Y TRABAJO FUTURO

En este artículo se describió la evaluación empírica del método de optimización de requisitos Web basado en la eficiencia de Pareto. El método le permite al diseñador evaluar entre las diferentes configuraciones de FRs que se podrán implementar mientras son optimizados los NFRs. La evaluación empírica se realizó para saber en qué casos el método de eficiencia de Pareto mejora o no la exactitud en las decisiones de diseño durante la etapa de requisitos, así como para saber si reduce o no el tiempo utilizado por los diseñadores de la aplicación Web. Los resultados demuestran que los diseñadores, utilizando la frontera de Pareto, obtuvieron un mejor marcador (efectividad) en poco tiempo (eficiencia) tratándose de modelos fáciles y complejos en comparación con los modelos *i**. Por lo tanto, concluimos que el método de Pareto, utilizado como complemento al modelado de requisitos Web con *i**, mejora la precisión y el tiempo necesario para la selección de entre diferentes configuraciones de FRs con el fin de obtener un equilibrio satisfactorio de NFRs (de acuerdo con su prioridad).

Como trabajo futuro, se replicarán los experimentos en otras condiciones para corroborar los resultados obtenidos. Además, se está trabajando en una representación gráfica de la tabla utilizada en la frontera de Pareto para realizar una evaluación empírica y compararla con los resultados obtenidos en estos experimentos.

AGRADECIMIENTOS

Este trabajo ha sido parcialmente financiado por los siguientes proyectos: Open.Mind (GV/2014/098) de la Generalitat Valenciana; PROFAPI2012/002 y PROFAPI2013/002 de la Universidad Autónoma de Sinaloa, México y MANTRA (GRE09-17) de la Universidad de Alicante.

REFERENCIAS

- [1] J. Verner, K. Cox, S. Bleistein, and N. Cerpa, "Requirements Engineering and Software Project Success: an industrial survey in Australia and the U.S.", *Australian Journal of Information Systems*, Vol. 13(1), 225-238, 2005.
- [2] C. Gupta, Y. Singh, D.S. Chauhan, "Dependency Based Process Model for Impact Analysis: A Requirement Engineering Perspective", *J. Comp. App.*, Vol. 6(6), 28-30, 2010.
- [3] D. Ameller, F. Gutiérrez, J. Cabot, "Dealing with Non-Functional Requirements in Model-Driven Development", in *Proc. 18th IEEE Int. Req. Eng. Conf.*, 2010, pp. 189-198.
- [4] I. Garrigós, J.N. Mazón, N. Koch, M.J. Escalona, J. Mylopoulos, Foreword of the *First International Workshop on the Web and Requirements Engineering*. IEEE, 2010, i-ii.
- [5] D. Bolchini, P. Paolini, "Goal-Driven Requirements Analysis for Hypermedia- Intensive Web Applications", *J. Req. Eng.* Vol. 9(2), 85-103, 2004.
- [6] J. Karlsson, K. Ryan, "A Cost-Value Approach for Prioritizing Requirements", *IEEE Software* Vol. (14), Nr. 5, 67 - 74, 1997.
- [7] A. Ginige, "Web Engineering: Managing the Complexity of Web Systems Development", in *Proc 14th Int. Conf. on Soft. Eng. and K. Eng.*, 2002, pp. 721-729.
- [8] E. Yu, "Towards Modeling and Reasoning Support for Early-Phase Requirements Engineering", in *Proc 3rd IEEE Int. Symp. on Req. Eng.*, 1997, pp. 226-235.
- [9] J.A. Aguilar, I. Garrigós, J.N. Mazón, J. Trujillo, "An MDA Approach for Goal- Oriented Requirement Analysis in Web Engineering", *J. Univ. Comp. Sc.* Vol. 16(17), 2475-2494, 2010.
- [10] J.A. Aguilar, I. Garrigós, J.N. Mazón, "A Goal-Oriented Approach for Optimizing Non-functional Requirements in Web Applications", in *Proc ER workshops: 8th Int. Work. on Web Inf. Syst. Mod.*, 2011, pp. 14-23.
- [11] J.A. Aguilar, I. Garrigós, S. Casteleyn, J.N. Mazón, "WebRED: A Model-Driven tool for Web Requirements Specification and Optimization", in *Proc 12th Int. Conf. on Web Eng.*, 2012, pp. 452-455.
- [12] F. Szidarovszky, M. Gershon, L. Duckstein, "Techniques for Multiobjective Decision Making in Systems Management", Elsevier Science Ltd, 574, 1986.
- [13] I. Garrigós, J.N. Mazón, J. Trujillo, "A Requirement Analysis Approach for Using *i** in Web Engineering", in *Proc 9th Int. Conf. on Web Eng.*, 2009, pp. 151-165.
- [14] M.J. Escalona, N. Koch, "Requirements Engineering for Web Applications - A Comparative Study", *J. Web Eng.* Vol. 2(3), 193-212, 2004.
- [15] B.J. Winer, D.R. Brown, K. M. Michels, "Statistical Principles in Experimental Design", McGraw-Hill, New York, 1991.
- [16] J.A. Aguilar, I. Garrigós, J.N. Mazón, J. Trujillo, "Web Engineering Approaches for Requirement Analysis- A Systematic Literature Review", in *Proc 6th Web Inf. Sys. and Tech.*, 2(2), 2010, pp. 187-190.
- [17] F. Molina, A. Toval, "Integrating Usability Requirements that can be Evaluated in Design Time into Model-Driven Engineering of Web Information Systems", *J. Adv. Eng. Softw.* Vol. 40, 1306-1317, 2009.
- [18] L. Chung B. Nixon E. Yu "Using Non-Functional Requirements to Systematically Select Among Alternatives in Architectural Design", in *Proc 1st Int. Work. on Arch. for Soft. Syst.*, 1995, pp. 31-43.
- [19] D.L. Moody, "Theoretical and practical issues in evaluating the quality of conceptual models: current state and future directions", *Data & Knowledge Engineering* Vol. 55, 243-276, 2005.
- [20] A. Anaby-Tavor, D. Amid, A. Fisher, H. Ossher, R.K.R. Bellamy, M. Callery, M. Desmond, S. Krasikov, R. Roth, T. Simmonds, J. de Vries, "An Empirical Study of Enterprise Conceptual Modeling", in *Proc Int. Conf. on Con. Mod.*, 2009, pp.55-69.

- [21] G. Shanks, E. Tansley, J. Nuredini, D. Tobin, R. Weber, "Representing part-whole relations in conceptual modeling: an empirical evaluation", *MIS Quarterly*, Vol. 32, Nr. 3, 553-573, 2008.
- [22] A. De Lucia, C. Gravino, R. Oliveto, G. Tortora, "An experimental comparison of ER and UML class diagrams for data modelling", *Empirical Software Engineering*, Vol. 15 Nr. 5, 455 - 492, 2010.
- [23] H. Al-Subaie T. Maibaum, "Evaluating the Effectiveness of a Goal-Oriented Requirements Engineering Method", in *Proc 4th Int. Work. on Comp. Eval. in Req. Eng.*, 2006, pp. 8-19.
- [24] M. Teruel, E. Navarro, V. López-Jaquero, F. Montero, P. González, "An Empirical Evaluation of Requirement Engineering Techniques for Collaborative Systems", in *Proc Int. Conf. on Eval. and Asses. in Soft. Eng.*, 2011, pp. 114-123.
- [25] B. Kitchenham, "DESMET: A method for evaluating Software Engineering methods and tools", *Computing & Control Engineering Journal*, Vol. 8(3), 120-126, 1997.
- [26] M. Morandini, A. Perini, A. Marchetto, "Empirical Evaluation of Tropos4AS Modelling", in *Proc 5th Int. i* Workshop*, 14-19, 2011.
- [27] H. Estrada, A. Martínez Rebollar, O. Pastor, J. Mylopoulos, "An Empirical Evaluation of the i* Framework in a Model-Based Software Generation Environment.", in *Proc Int. Conf. of Ad. Inf. Syst. Eng.*, 2006, pp. 513-527.



José Zubcoff es profesor contratado doctor en el Departamento de Ciencias del Mar y Biología Aplicada de la Universidad de Alicante. Obtuvo su doctorado en informática en la Universidad de Alicante siendo sus líneas de investigación la minería de datos, inteligencia de negocio, gestión de datos abiertos y los métodos estadísticos. Ha publicado trabajos en varias revistas de impacto como DKE, ISOFT,

Autoimmunity, and Desalination, así como conferencias internacionales como DAWAK, Data Mining, GeoInformatic, etc.



Irene Garrigós es profesora contratada doctora en la Universidad de Alicante en la cual ha defendido su tesis en Ingeniería Informática. Perteneció al grupo de investigación Lucentia de esa misma universidad. Tiene en su haber publicaciones en revistas, conferencias y workshops especializados de carácter internacional (como ICWE, ER, WISE, JISBD, ISOFT, JWE etc), participando

también como parte del comité científico en algunos de estos eventos. Ha realizado estancias de investigación en Bélgica (Vrije Universiteit Brussel) y Holanda (Technische Universiteit Eindhoven). Su investigación se centra principalmente en ingeniería Web, personalización, sistemas adaptativos, desarrollo dirigido por modelos e ingeniería de requisitos. Ha participado en la organización de varias ediciones de los workshops BEWEB y WeRE..



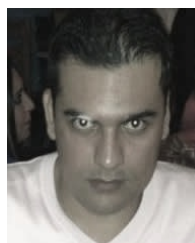
Sven Casteleyn obtuvo su Master (1999) en Ciencias de la Computación y su doctorado en Ciencias (2005) de la Vrije Universiteit Brussel, Bélgica. Después de su doctorado, trabajó sucesivamente como investigador post-doctoral y profesor en la Vrije Universiteit Brussel, Bélgica, la

Universidad Politécnica de Valencia (beca Marie Curie IEF), y la Universidad Jaume I en Castellón, España. Su investigación se centra en los campos de la ciencia de la Web, Ingeniería Web, Web Semántica y computación de móviles. Ha publicado dos libros, dos capítulos de libros, y más de 40 artículos en revistas internacionales, conferencias y talleres. Ha sido Program Chair del International Conference on Web Engineering 2014, track y demo chair de varios otras conferencias, y PC member de 50+ conferencias y talleres. Tiene un h-index de 17.



Jose-Norberto Mazón es profesor ayudante doctor y miembro del grupo de investigación WaKe de la Universidad de Alicante. Sus líneas de investigación abarcan temas relacionados con la ingeniería de datos, como gestión de datos abiertos, calidad de datos, procesos de integración de datos, soluciones de inteligencia de negocio en escenarios "big data", etc. Ha publicado sus trabajos en

revistas científicas de impacto internacional como Decision Support Systems, Information Sciences, Data & Knowledge Engineering o ACM Transaction on the Web. Por otra parte, ha realizado ponencias en diversos congresos internacionales y nacionales dentro del ámbito de la ingeniería del software y las bases de datos. Además, Jose Norberto forma parte del equipo de desarrollo del proyecto de apertura de datos de la Universidad de Alicante impulsado por el Vicerrectorado de Tecnologías de la Información, incluyendo su portal de datos abiertos <http://datos.ua.es>. Puedes contactar con él a través de Twitter en @jnmazon



José Alfonso Aguilar obtuvo su Maestría en Informática Aplicada por la Universidad Autónoma de Sinaloa, México en 2007 y recibió el título de Doctor en Aplicaciones de la Informática por la Universidad de Alicante, España en 2011. De 2007 a 2009 se desempeñó como desarrollador de software y Asesor de Calidad de Proyectos, Procesos y

Productos en el Departamento de Sistemas Computación de Coppel. Desde 2012 ha sido Profesor Investigador de Tiempo Completo en el grupo de investigación Señales y Sistemas de la Facultad de Informática Mazatlán en la Universidad Autónoma de Sinaloa, México. Ha publicado varios artículos en conferencias, talleres y revistas como ICWE, WEBIST, ER, ICCSA, WISM, IEEE Latin America Transactions y JUCS entre otros. Sus áreas de investigación incluyen la Ingeniería Dirigida por Modelos, Ingeniería de Requisitos, Ingeniería Web y Modelado de Sistemas. Aguilar fue galardonado con el Mejor Demostración de Software en la 12ª International Conference on Web Engineering (ICWE) en 2012 por la herramienta WebREd-Tool.